

# Algoritmos y Estructuras de Datos II

## Tercer parcial

Sábado 5 de junio de 2021

### Aclaraciones

- El parcial es individual en todas sus etapas, y a libro abierto.
- Se responderán consultas por Discord, sólo de interpretación del enunciado, a través del canal ‘Mesa de Docentes’ de la categoría PARCIALES.
- Cualquier aclaración sobre el enunciado se realizará a través del canal ‘anuncios-evaluaciones’.
- El parcial durará 3 horas, de 9 a 12hs, y la entrega se realizará a través del Campus hasta esa hora, en forma estricta. Se desestimarán entregas fuera de tiempo, sin excepciones.
- Cada ejercicio se calificará con **Perfecto**, **Aprobado**, **Regular**, o **Insuficiente**.
- Para aprobar la cursada se deben tener todos los ejercicios aprobados de todos los parciales.
- Los ejercicios con **R** o **I** se recuperarán por separado al final del cuatrimestre.

### Ej. 1. Elección de estructuras

Nos encargaron el diseño del módulo Fotónica, descrito en el TAD de más abajo, el cual permite la recuperación de fotos de un usuario. Fotónica se encargará de registrar los datos de las fotos, de las cuales sólo tiene su id numérico ya que están almacenadas en otro sistema, y permitirá recuperarlas de acuerdo a distintos criterios de búsqueda. Cada foto puede tener cero o más etiquetas asociadas (en inglés, *tags*), las cuales son palabras que permiten caracterizar a la foto, y que por ahora sólo se sabe que tendrán un límite máximo de caracteres, a definir posteriormente, además de una fecha de creación (día, mes y año) y la ubicación de donde se tomó una foto, que se representa por medio de un solo número natural.

**TAD FOTO** es NAT  
**TAD TAG** es String  
**TAD UBICACIÓN** es NAT

**TAD FOTÓNICA**

**géneros**            fotónica

**observadores básicos**

fotos : fotónica  $\rightarrow$  conj(foto)

tagsPorFoto : fotónica  $s \times$  foto  $f \rightarrow$  conj(tag)  $\{ \_?fotoRegistrada?(s, f) \}$

fechaFoto : fotónica  $s \times$  foto  $f \rightarrow$  nat  $\times$  nat  $\times$  nat  $\{ \_?fotoRegistrada?(s, f) \}$

ubicaciónFoto : fotónica  $s \times$  foto  $f \rightarrow$  ubicación  $\{ \_?fotoRegistrada?(s, f) \}$

**generadores**

iniciar :  $\rightarrow$  fotónica

agregarFoto : fotónica  $s \times$  foto  $f \times$  conj(tag)  $\times$  nat  $d \times$  nat  $m \times$  nat  $a \times$  ubicación  $\rightarrow$  fotónica  $\{ \neg \_?fotoRegistrada?(s, f) \}$

borrarFoto : fotónica  $s \times$  foto  $f \rightarrow$  fotónica  $\{ \_?fotoRegistrada?(s, f) \}$

**otras operaciones**

$\_?fotoRegistrada?$  : fotónica  $\times$  foto  $\rightarrow$  bool

fotosPorTag : fotónica  $\times$  tag  $\rightarrow$  conj(foto)

fotosDelDía : fotónica  $\times$  nat  $\times$  nat  $\rightarrow$  conj(foto)

fotosPorUbicación : fotónica  $\times$  ubicación  $\rightarrow$  conj(foto)

ubicaciónMásFotografiada : fotónica  $s \rightarrow$  ubicación  $\{ \neg \emptyset?(fotos(s)) \}$

**Fin TAD**

Dadas las siguientes operaciones con las complejidades temporales de peor caso indicadas (donde  $n$  es la cantidad de fotos actuales,  $t$  es la cantidad de tags de una foto que se agrega y  $ts$  la cantidad de tags distintos actuales del sistema):

- **AGREGARFOTO**(**inout** sistema: fotónica, **in** id: foto, **in** tags: secuencia(tag), **in** día: nat, **in** mes: nat, **in** año: nat, **in** lugar: ubicación)  
 {**Pre**: La foto 'id' no está registrada. La fecha indicada en mes, día y año es válida.}  
 Se agrega una foto al sistema con sus tags, así como el día, mes y año de la foto.  
Complejidad:  $O(\log(n) + t)$
- **BORRARFOTO**(**inout** sistema: fotónica, **in** id: foto)  
 {**Pre**: La foto 'id' ya está registrada}  
 Elimina el registro de una foto en el sistema.  
Complejidad:  $O(\log(n) + ts)$
- **¿FOTOREGISTRADA?**(**in** sistema: fotónica, **in** id: foto, **out** res: bool)  
 Indica si la foto indicada está o no registrada en el sistema.  
Complejidad:  $O(\log(n))$
- **TAGSPORFOTO**(**in** sistema: fotónica, **in** id: foto, **out** res: secuencia(tag))  
 {**Pre**: La foto 'id' ya está registrada}  
 Obtiene los tags que tiene una foto determinada.  
Complejidad:  $O(\log(n))$
- **FOTOSPORTAG**(**in** sistema: fotónica, **in** tag: tag, **out** res: secuencia(foto))  
 Obtiene las fotos que tienen un tag determinado.  
Complejidad:  $O(1)$
- **FOTOSDELDÍA**(**in** sistema: fotónica, **in** día: nat, **in** mes: nat, **out** res: secuencia(foto))  
 Dado un día y mes, obtiene todas las fotos de esa fecha, sin importar el año.  
Complejidad:  $O(1)$

Se pide:

1. Plantear la estructura de representación del módulo FOTÓNICA, que provee las operaciones mencionadas más arriba. Se debe explicar detalladamente qué información se guarda en cada parte, las relaciones entre las partes, cómo se evitaría que haya incongruencias en caso de haber datos redundantes, y las estructuras de datos subyacentes.
2. Resumir de qué manera se resolvería cada una de las operaciones listadas anteriormente, de acuerdo la estructura de representación elegida, y haciendo las aclaraciones sobre *aliasing* necesarias.
3. Escribir el algoritmo de la operación agregarFoto, y justificar detalladamente su complejidad.
4. (**Opcional, por puntaje extra**) Indicar qué se debería modificar/agregar a la estructura definida anteriormente y cómo cambiarían los costos de las otras operaciones, en caso de verse afectadas, para poder implementar la operación indicada a continuación:
  - **UBICACIÓNMÁSFOTOGRAFIADA**(**in** sistema: fotónica, **out** res: ubicación)  
 {**Pre**: Hay al menos una foto registrada}  
 Se desea saber cuál es la ubicación que tiene más fotografías en el sistema.  
Complejidad:  $O(\log(u))$ , donde  $u$  es la cantidad de ubicaciones distintas de las fotos registradas en el sistema)

Para la resolución del ejercicio no está permitido utilizar módulos implementados con tabla hash de base. Esto se debe a dos motivos: uno porque queremos que combinen el resto de las estructuras vistas, y otro porque los peores casos de la tabla hash exceden los que se piden en los ejercicios (por ejemplo, duplicar el espacio en una tabla cuesta  $O(n)$  en el peor caso).

Donde se menciona **secuencia** en las definiciones de las operaciones anteriores se debe usar alguno de los módulos vistos que se explican con SECUENCIA, de acuerdo a lo que está en el apunte de Módulos Básicos, o definir uno nuevo que también se explique con ese TAD.